THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 59, No. 2, February 1980
Printed in U.S.A.

B.S.T.J. BRIEF

Serial Coding for Cyclic Block Codes

By S. V. AHAMED

(Manuscript received August 2, 1979)

I. INTRODUCTION

In 1972 the concept of serial encoding and decoding for single error correcting BCH codes was introduced.[1,2] In this note, the concept of serial encoding and decoding is generalized and the timing diagrams are presented for a typical (n, k) cyclic block code. The implementation in conventional technology uses only one exclusive-OR gate and is presented for all $(n - k)$ order generator polynomials for any code n bits long. The implementations presented are valid for all cyclic block encoders and for all decoders with single error correction with multiple error detection capability.

Most of the literature[3,4] on encoders and decoders emphasizes the binary division process between the data polynomial $d(X)$, k bits long, and the generator polynomial $g(X)$, p bits long, in conventional BCH type of error correction. In general, the division is accomplished by distributing a series of exclusive-OR gates embedded in a shift register. The associated logical functions to be accomplished to maintain synchronism between bits of the code word $c(X)$ at the encoder and the corresponding synchronizing logic at the decoder are both ignored.

With the recent advent of high-speed logic circuitry, it appears redundant to use a large number of exclusive-OR gates and unduly complicate the logic involved. Instead, a single exclusive-OR gate* may be used in the serial coding where the contents of the shift register in the encoders and decoders are completely circulated once for each step

* The principle of performing serial encoding by using a single exclusive-OR gate has been described in Refs. 1 and 2. It is the object of this note only to extend the basic concept to all (n, k) cyclic codes and to present implementational details of the actual codecs.

of the division, instead of shifting once for each step as in conventional dividers. Three distinct advantages accrue from this type of serial encoding and decoding:

(*i*) The number of gates is considerably reduced, especially if the generator polynomial is densely distributed.

(*ii*) The generator polynomial may be changed from block to block by changing the bit patterns of the circulating register, $g'(X)$ (i.e., $g(X)$ without the leading high order bit).

(*iii*) The synchronization is most easily achieved between (*a*) the bits in $d(X)$ and in $c(X)$ at the encoder and (*b*) the bits in the received polynomial $R(X)$ and corrected data stream $d(X)$ (at its original rate) at the decoder.

II. DIFFERENCES BETWEEN SERIAL CODING AND CONVENTIONAL CODING

Conventionally, each step of division between $d(X)$ and $g(X)$ is achieved by a feedback of the exclusive-OR summation of the paired remainder and the incoming bit into a series of exclusive-OR gates spatially distributed in a shift register. Now consider two registers, $g'(X)$ and SR (Fig. 1), circulating synchronously, which feed into a single exclusive-OR gate.* The circulation time is chosen to equal the time interval between the data bits of $d(X)$. After each circulation, the leading bit is fed into X_P†, and the arriving data bit occupies X_O. The gate, S_A, responds to the content of X_P, closing only if it is "one."‡ After k such steps of division, $(n - k)$ parity bits would be left in SR. Whereas all the exclusive-OR gates in the conventional dividers operate simultaneously, the one single exclusive-OR gate in a serial divider acts sequentially, but at a much higher rate.

III. IMPLEMENTATION OF ENCODERS

The arrival rate of data is 1 bit every nt seconds (see Fig. 2). The first p bits of data are accommodated in an interim data store, R_i, with S_1 and S_2 open. Then during the next nt seconds, the first p bits of data are moved from R_i to R_2 and SR, while SR empties the parity bits of the previous data block to R_1 via switch S_3. The $(p + 1)$st bit moves into X_O via S_1, and high-order bit of data into X_P via S_5. The contents of SR and $g'(X)$ are circulated once via S_3 during the next nt seconds,

* SR is the syndrome register.

† $p = (n - k)$ = number of parity bits in a n-bit code word with k data bits.

‡ A simple AND gate will accomplish the necessary function.

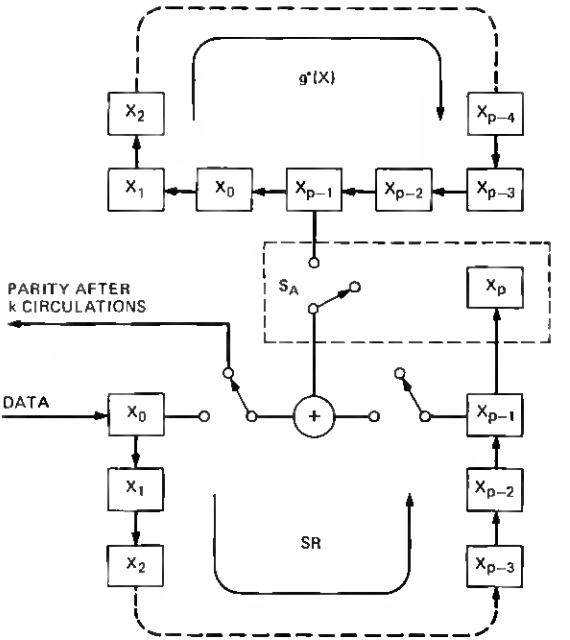

Fig. 1—Serial shift register for a k-bit data block with p parity bits $g'(X) = g(X) + X^p$.

thus performing one step of the k step division[1] cycle. After $(k - p)$ such steps, the first data block is completely received and, for the next $(p \cdot nt)$ seconds, the first p bits of the next data block accumulate in R_i. Meanwhile, SR and $g'(X)$ complete the last p steps of the division process for the current data block. Data synchronization is achieved by R_1, R_2, and R_3 via S_6, S_7, and S_8. Register R_1 receives the parity bits in nt seconds and empties it uniformly every kt seconds. R_2 receives the first p bits of every data block during nt seconds simultaneously with SR, and empties it one bit every kt seconds. R_3 receives the last $(k - p)$ bits of data, one every nt seconds, and empties it one every kt seconds. The timing diagram is shown in Fig. 3. A scrutiny of Fig. 3 now raises questions as to the adequacy of the data register arrangement shown in Fig. 2 when p becomes a sufficiently small fraction of $(p < n/3)$.* Under such circumstances, R_3 essentially has to be partitioned into segments. The new configuration can then be designed to satisfy both the data synchronism requirement and the shift register stipulation that each of these stores be shifting in or shifting out only during a predefined period within the encoding cycle.

* The limiting value of the fraction is achieved when $T = nkt = pt(2k + n)$ and the register R_3 shifts out immediately after it has shifted in, thus leading to $p \geq 0.414\ k$.

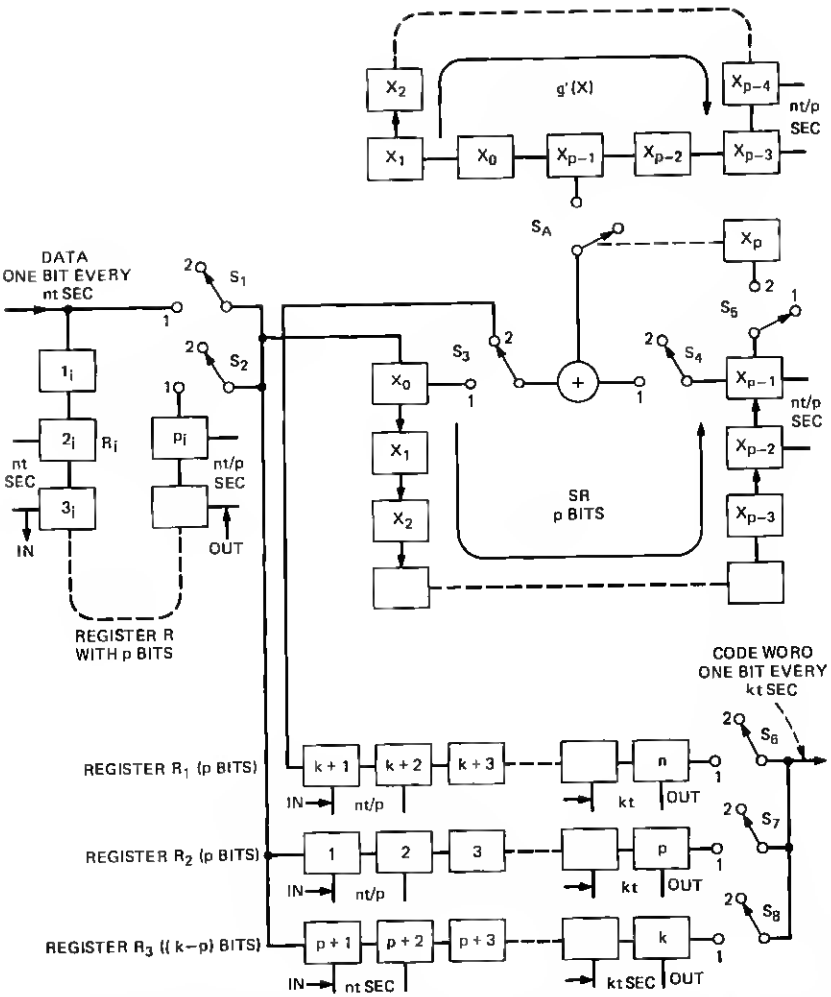

Fig. 2—Serial encoder for a k-bit data block with p parity bits. The generator polynomial is $g(X)$ of the order p and $g'(X) = g(X) + X^p$. The code word has $n = (k + p)$ bits and is transmitted uniformly every T seconds. $t = (T)/(nk)$ seconds.

IV. IMPLEMENTATION OF DECODERS

Decoding consists of three distinct steps: (*i*) The calculation of the syndrome. (*ii*) Shift and divide procedure for the syndrome. (*iii*) Comparison of the remainder with $\Gamma(X)$ calculated and stored as $(X^{n-1}/g(X))$.

Thus the decoder shown in Fig. 4 has two syndrome registers SR_1 and SR_2. While SR_1 is calculating the syndrome of one data block, SR_2 is shifting and dividing for the previous data block. The comparison

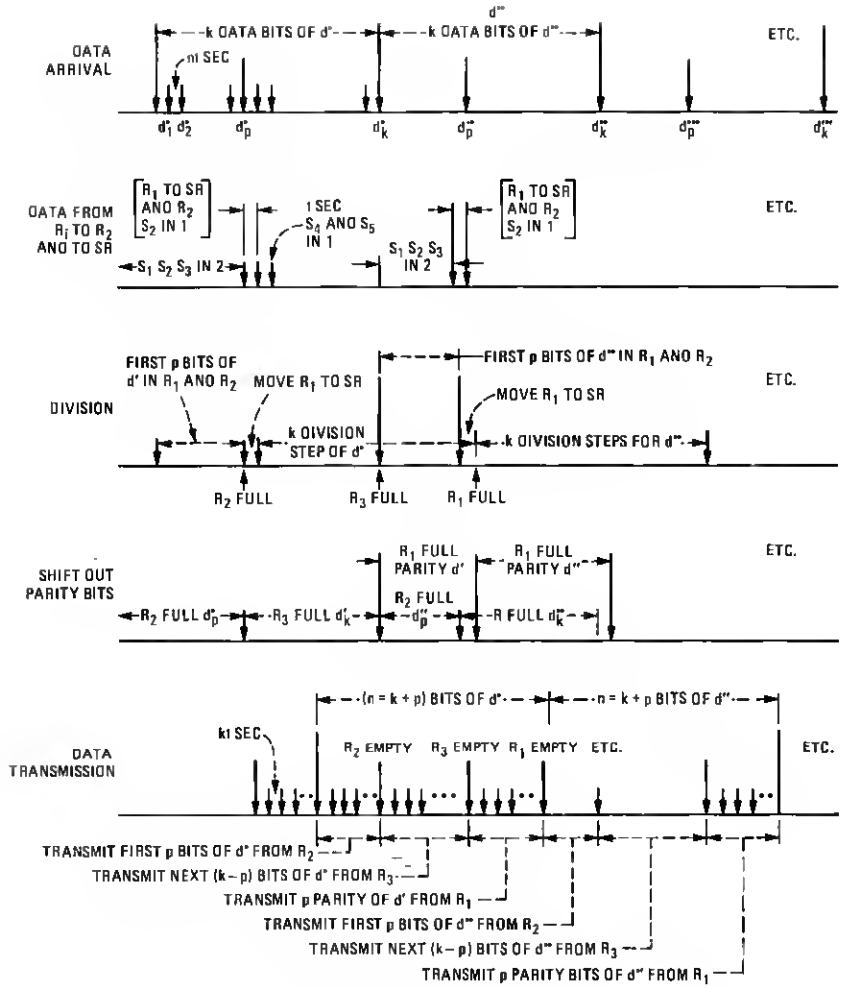

Fig. 3—Timing diagram for (n, k) BCH code encoder.

with $\Gamma(X)$ is accomplished by an exclusive-OR gate located between shift register $\Gamma(X)$ and SR. When all the p bits of SR after the lth shift and divide operation of SR match all the p bits of $\Gamma(X)$, then the lth bit is complemented for error correction at the output exclusive-OR gate g_5. After comparison of the contents of SR and of $\Gamma(X)$, the signal is inverted at I, and the toggle T closes only if all the p bits received by it are ones. Data synchronization is achieved by S_1, S_2, S_3, and S_4. These switches alternate between positions 1 and 2 for consecutive data blocks. D_1 and D_2, each k bits long, shift in every pt seconds and shift out every nt seconds. The timing diagram for the decoder is shown in Fig. 5.

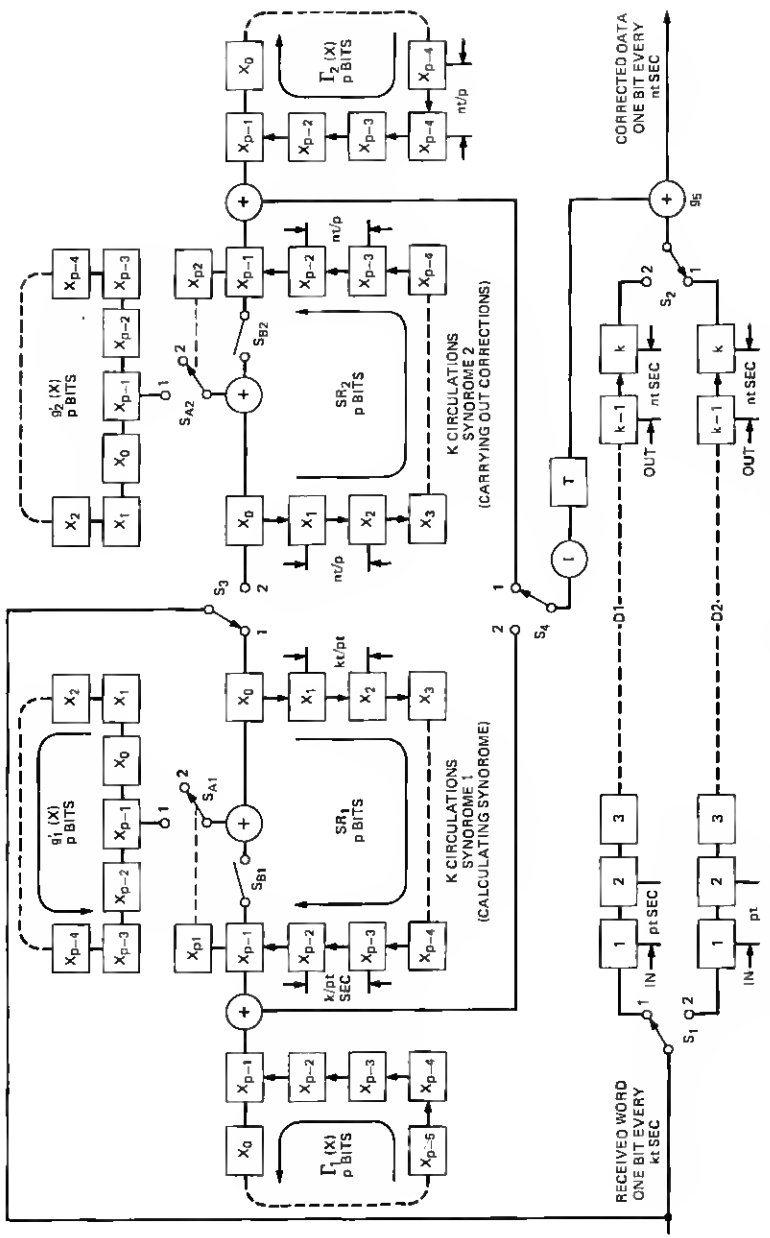

Fig. 4—Decoder circuit for (n, k) single error correcting BCH code, $r(X) = (X^{n-1}/g(X))$.

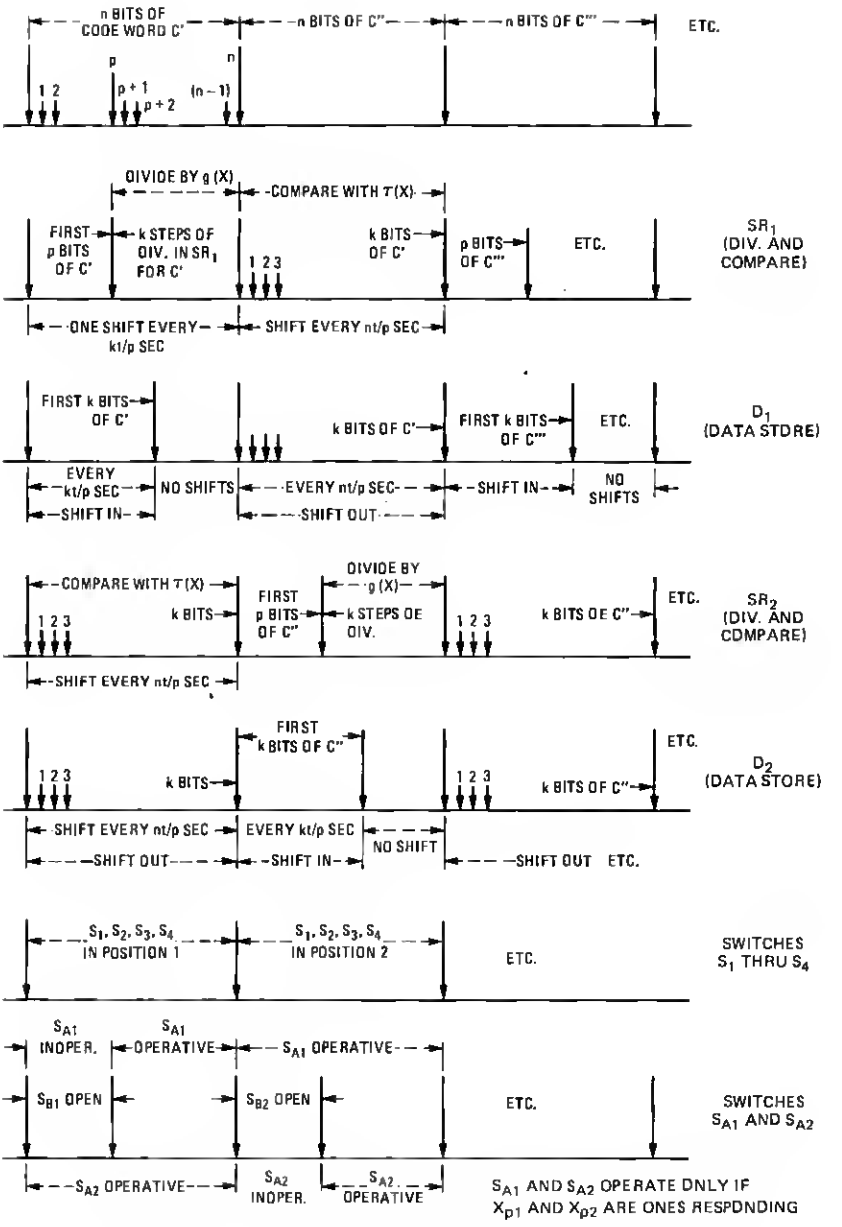

Fig. 5—Timing diagram for the decoder shown in Fig. 4.

V. CONCLUSIONS

Serial codecs accomplish single error corrections with fewer exclusive-OR gates than conventional coders. The generator polynomial of serial codecs may be changed from one block to the next by changing the bits in $g'(X)$ and in $\Gamma(X)$. When secrecy in coding is required, this would offer a distinct advantage. When the data rate for the message channel is known to be at b bits per second, then a single clock at $(b \times k)$ Hz can achieve perfect synchronization between the incoming and outgoing data of the serial codecs. Further, if the number of parity bits p is an integer fraction of the number of data bits k, then all the shift registers in the codecs may be shifted by a clock derived as an integer fraction of the master clock at $(b \times k)$ Hz.

VI. ACKNOWLEDGMENT

The author acknowledges the contribution of the B.S.T.J. reviewer who pointed out that these configurations are valid for encoding all cyclic block codes and for decoders for single error correction with multiple error detection capability. A slight modification of the syndrome vs $\Gamma(X)$ comparator would be necessary for an automatic request for retransmission of the erroneous block.

REFERENCES

1. S. V. Ahamed, "The Design and Embodiment of Magnetic Domain Encoders and Single-Error Correcting Decoder for Cyclic Block Codes," B.S.T.J., *51*, No. 2 (February 1972), pp. 461–485.
2. S. V. Ahamed, "Extension of Multidimensional Polynomial Algebra to Domain Circuits with Multiple Propagation Velocities," B.S.T.J., *51*, No. 8 (October 1972), pp. 1919–1922.
3. W. W. Peterson and E. J. Weldon. *Error-Correcting Codes*, Cambridge Mass.: MIT Press, 1972.
4. E. R. Berlekamp, *Algebraic Coding Theory*, New York: McGraw-Hill, 1968.